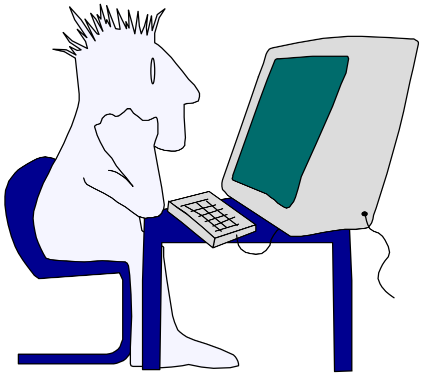
Additional Information:
SOFTWARE QUALITY



OUTLINE

- Software product assurance functions
- Specifications requirements.
- Required program documents.
- Other Considerations

SOFTWARE PRODUCT ASSURANCE FUNCTIONS

- Hazard Analysis
  - Preliminary Hazard Analysis
  - System/Subsystem Hazard Analysis
  - Operation and Support Hazard Analysis
- Reliability Model/Prediction
- Software Configuration Management
  - Software Configuration Identification
  - Software Change Control
- Software Design and Inspection Requirements

SOFTWARE PRODUCT ASSURANCE (con't)

- SW Nonconformance Reporting & Corrective Action.
- Failure Review Board (also analyzing SW problems).
- Software audits & surveys.
- Software reviews.
- Software testing (test procedures & reports).
- Design reviews.
- Software documentation standards & record.
- Formal bug and error reporting.
- Timing issues and sizing budgets.

SPECIFICATIONS REQUIREMENTS

- **SPECIFICATIONS THAT ARE REVIEWABLE**
- **SPECIFICATIONS THAT ARE USABLE BY DESIGNERS AND BY RELIABILITY/SAFETY**
- **SPECIFICATIONS THAT ARE FORMALLY ANALYZABLE.**
  - **Completeness and robustness checks.**
  - **Safety analysis**
  - **Simulation**
  - **Standard system engineering analysis.**
- **SPECIFICATIONS WHICH CAN GENERATE TEST DATA.**

SPECIFICATIONS REQUIREMENTS: PROGRAMMING STANDARDS

- **Structured Programming**
  - **Well defined design approach**
  - **Extensive commenting**
  - **No “clever” programs**
- **Modularity**
- **Use of CASE Tools**
- **Standard formats and nomenclature**
- **Language standard**
- **Standard compilers and platforms.**

SPECIFICATIONS REQUIREMENTS: Required Program Documents

- **SYSTEM SAFETY PROGRAM PLAN (SSPP)**
- **SOFTWARE STANDARDS & PROCEDURES MANUAL**
- **SOFTWARE DEVELOPMENT PLAN (SDP)**
- **SOFTWARE REQUIREMENTS SPECIFICATIONS (SRS)**
- **SOFTWARE TEST PLAN AND PROCEDURES (STP/STPR)**
- **DATA BASE DESIGN DOCUMENT**

OTHER CONSIDERATIONS

- Hardware/software interfaces.
- Computer controlled servo-actuators.
- RF noise problems.
- Electronic component reliability.
  - Harden components to EMI.
  - Isolate components from launch vibration.
  - Sensors used in a SW control system must be robust.
  - For critical systems use proven hardware and technology.

Center Software Assurance Activities

- Production quality metrics
- Software inspection training
- Software inspection
- Code “walk-thru”
- V&V
- IV&V

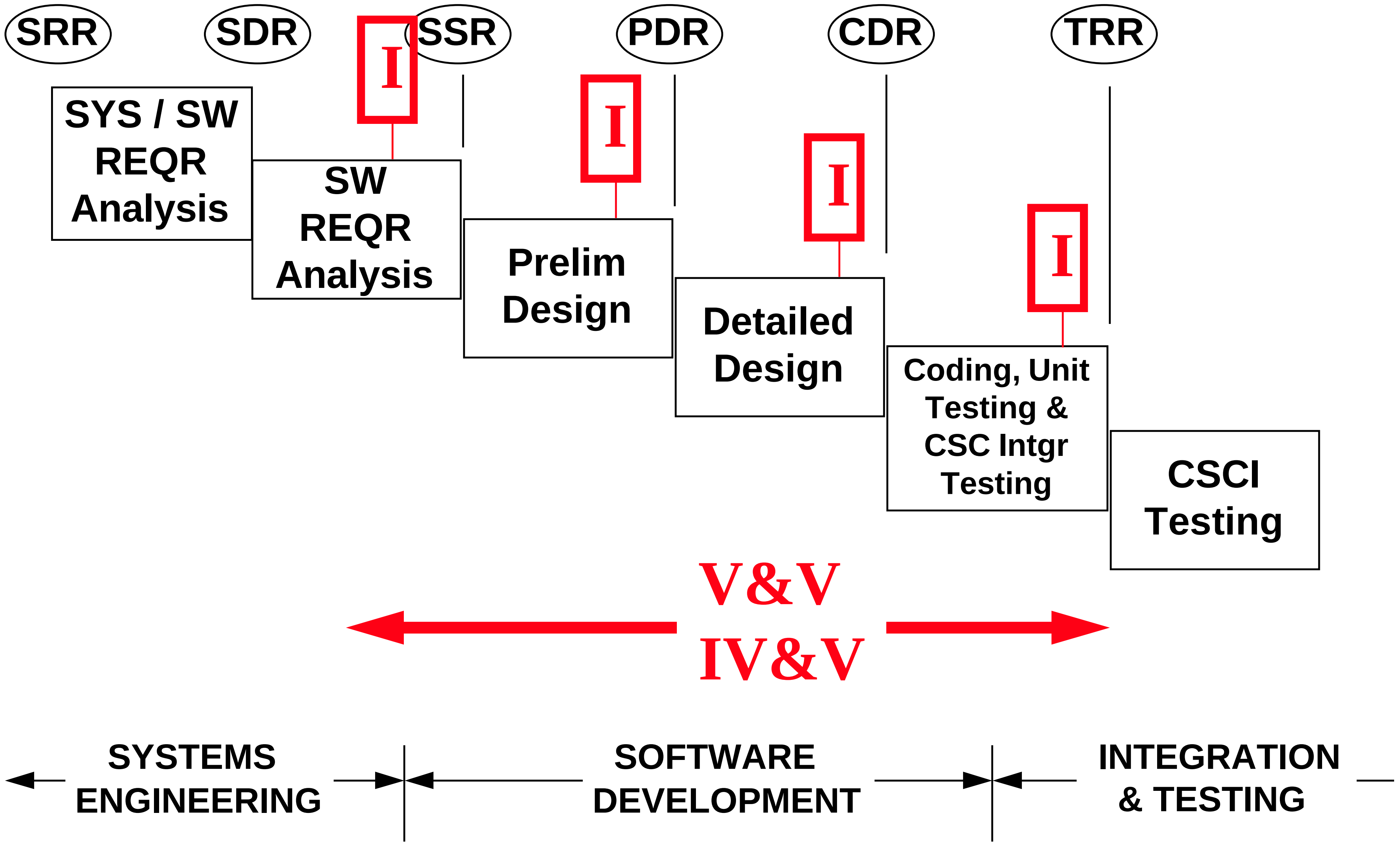
Benefits of Formal Inspection

- Objective is to remove defects as early as possible in the development process.
- Structured, well defined review process for finding and fixing defects.
- Metrics and checklists used to improve quality.
- Follows TQM techniques--working together as a team.
- Responsibility for work product shared by author's peers.
- Supported by NASA & DOD Standards.

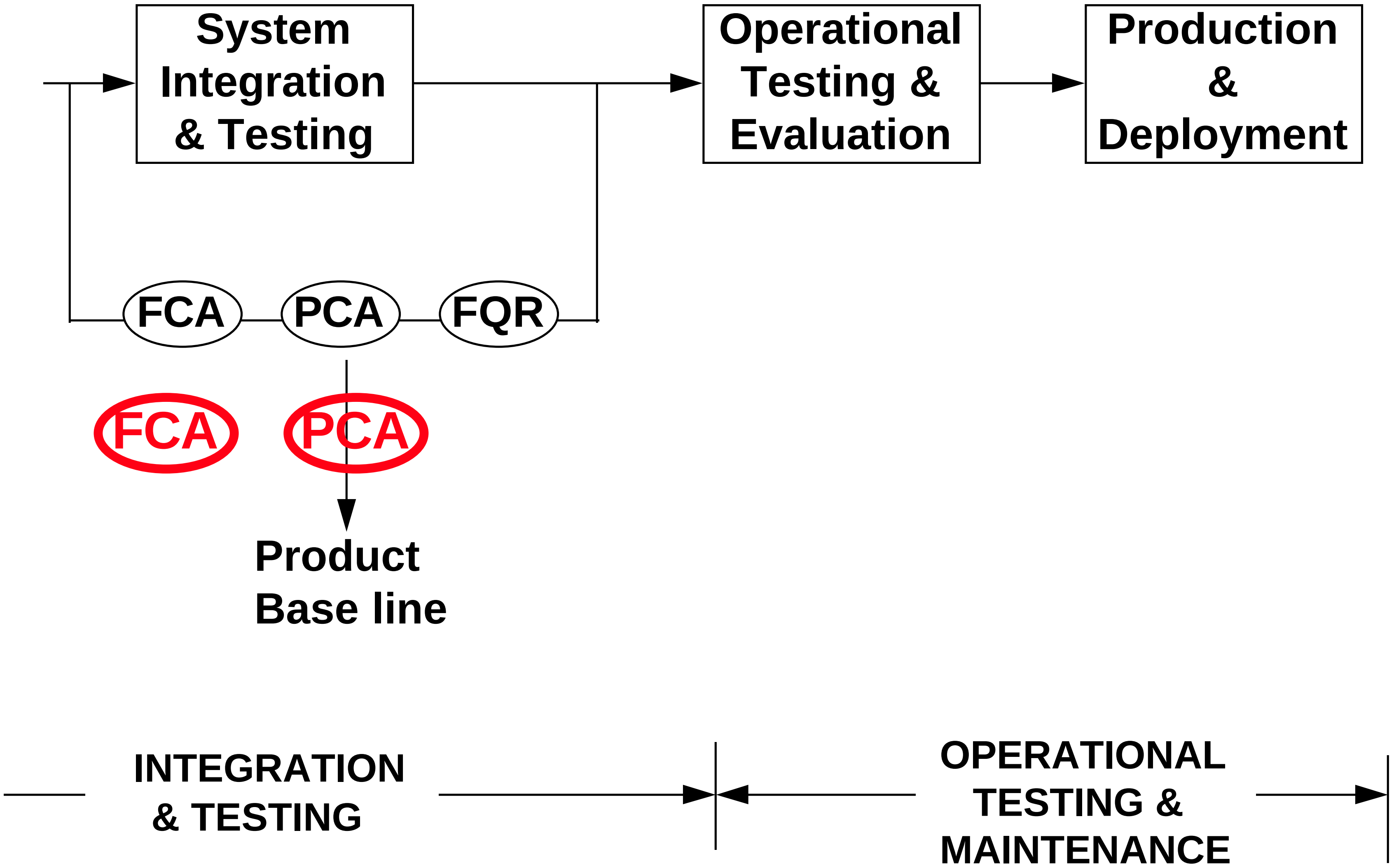
Formal Inspection Activities (Typ.)

- Implementation of requirements.
- Not implementing code (where no requirements exist).
- Review of pseudo code.
- Review of mechanics.
- Review of data structure.

SW Development Process



SW Development Process



System Integration & Testing

Operational Testing & Evaluation

Production & Deployment

FCA

PCA

FQR

FCA

PCA

Product Base line

INTEGRATION & TESTING

OPERATIONAL TESTING & MAINTENANCE

T-inspect sheets (opt)

Some Additional Software Recommendations for Major Projects

from a Review of a Major SW Program

- Provide consistent SW development coding guidelines among contractors.
- V&V inspections by contractors should pay close attention to off-nominal cases (crew/ground error, H/W failure, SW error conditions).
- V&V inspection should (1) focus on verifying consistency of two levels of descriptions for modules (2) consistence between modules requirements and design platform and (3) correctness wrt H/W and SW platforms.
- Provide for independence of IV&V.
- Safety standards and guidelines should include real SW standards.

Some Additional Software Recommendations for Major Projects

from a Review of a Major SW Program

- Coordination between system-safety program and SW activities. Create a safety plan and hazard analysis.
- Have sufficient personnel at SR&QA offices at centers to support software-related activities.
- Provide sufficient oversight and evaluation of SW development activities by the center SR&QA offices.
- Provide for multiple centers on the same program having and enforcing the same standards & procedures.
- Have a well documented maintenance & upgrade process.
- Provide for visibility for potential SW problems by defining in detail requirements to report SW reliability, QA or safety problems to the program-level safety organization.

Some Additional Software Recommendations for Major Projects

from a Review of a Major SW Program

- Assign all functions within the flight software process to a specific NASA or contractor organization rather than the “flight software community.”
- Provide accepted policies and guidelines for development & implementation of SW V&V, IV&V, reliability, QA & safety.
- Provide sufficient resources, personnel and expertise devoted to developing them.
- Provide sufficient resources, manpower & authority to compel development contractors to provide sufficient information to assure proper processes are followed.

Some Additional Software Recommendations for Major Projects

from a Review of a Major SW Program

- Capture lessons learned in the development, maintenance, and assurance of software to be used by other programs.
- Precisely identify the information that each development and oversight contractor is responsible for making available to each other and to the community as a whole.
- Put in place mechanisms necessary to ensure that programs are given all information needed to make intelligent implementations of SW oversight functions.

>P14.1 (opt)

Conclusions

- There are a number of software product assurance activities including formal inspection, production quality metrics, software inspection training, code “walk-thru,” V&V and IV&V which greatly enhance software quality.
- There are a number of documentation standards (including the SSPP, SOFTWARE STANDARDS & PROCEDURES MANUAL, SDP, SRS, STP/STPR etc.) which if used properly increase SW quality.
- LeRC is currently having great success in applying many of the above software quality tools. This is especially true of formal inspections at various stages of the SW design process.
- Major SW programs have also yielded excellent recommendations to improve SW quality of large projects and programs.

END

ADDITIONAL INFORMATION

PROBLEM: Chemical Reaction Microgravity Experiment -- WHAT *ELSE* CAN GO WRONG?

- Given: The Chemical Reaction Microgravity Experiment.
- Convert to total computer control with a single processor (eliminate individual PLC controllers). This will allow operation from ground and free astronauts to do other tasks.
- Also add carrousels to automatically change gas cartridges and control this with the computer as well.
- PERFORM A FAILURE MODES AND EFFECTS ANALYSIS
- PAY PARTICULAR ATTENTION TO SW & H/W Problems.
- HOW CAN SOFTWARE PROBLEMS BE AVOIDED?

F; P14-1 p.1

COMPUTER CONTROLLED Chemical Reaction Microgravity Experiment

- SENSORS (digital output sensors)
  - Pressure transducer (requires 12 volt excitation).
  - Temperature Sensor (thermocouple).
  - Flow meters
- COMPUTER CONTROLLED DEVICES:
  - Valves (control feedstock and product outflow).
  - Heating and cooling of the pressure vessel.
  - Remove ball valves and install additional solenoid valves.
  - Install two carrousels for experimental gases.

F; P14-1 p.2

COMPUTER CONTROLLED Chemical Reaction Microgravity Experiment (con't)

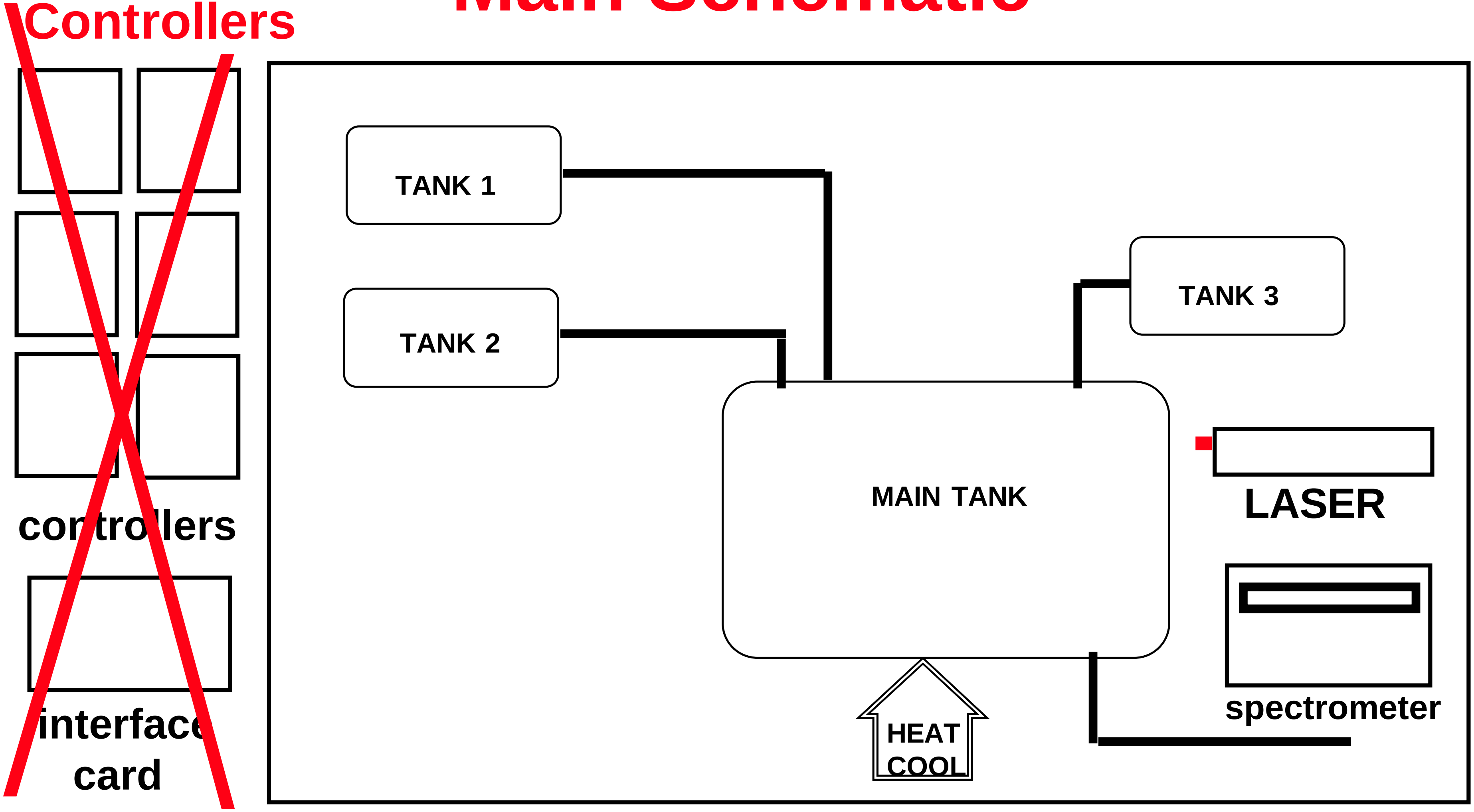
- COMPUTER:
  - Input /Output (I/O) Boards : cards which receive analog signals and convert to digital signals for computer and/or take computers digital signal and convert it to output signal to valves and heating and cooling elements).
  - Computer Microprocessor
  - Computer Disk Drives
  - Computer Power Supply
- CABLING (provide signal and control paths).

F; P14-1 p.3

CHEMICAL REACTION MICROGRAVITY EXPERIMENT

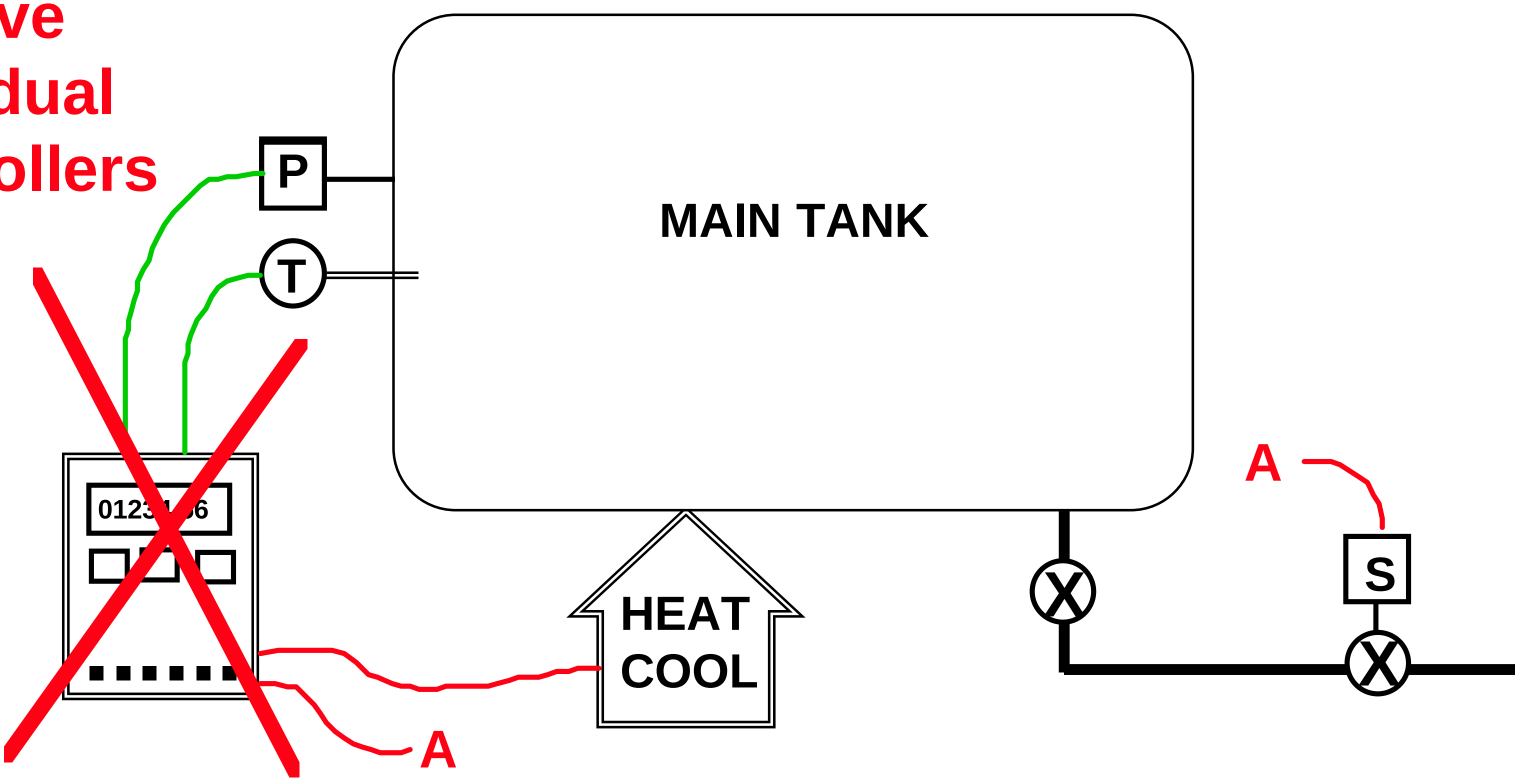
Main Schematic

Remove Controllers



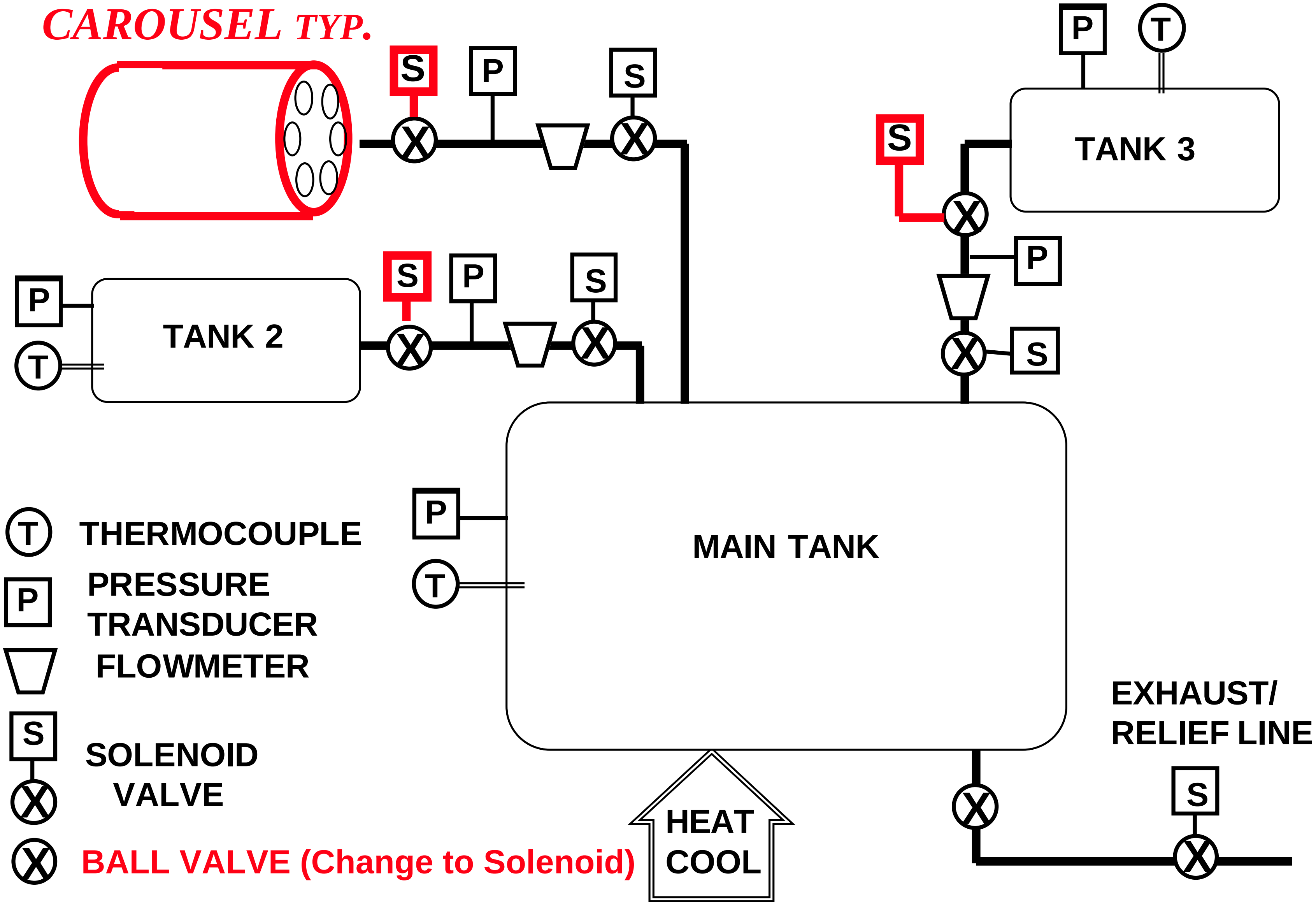
MAIN TANK THERMAL CONTROL

- Controller fixes tank temperature and controls “safety relief valve.”
- Heating and cooling is done by electric heaters or refrigeration units.

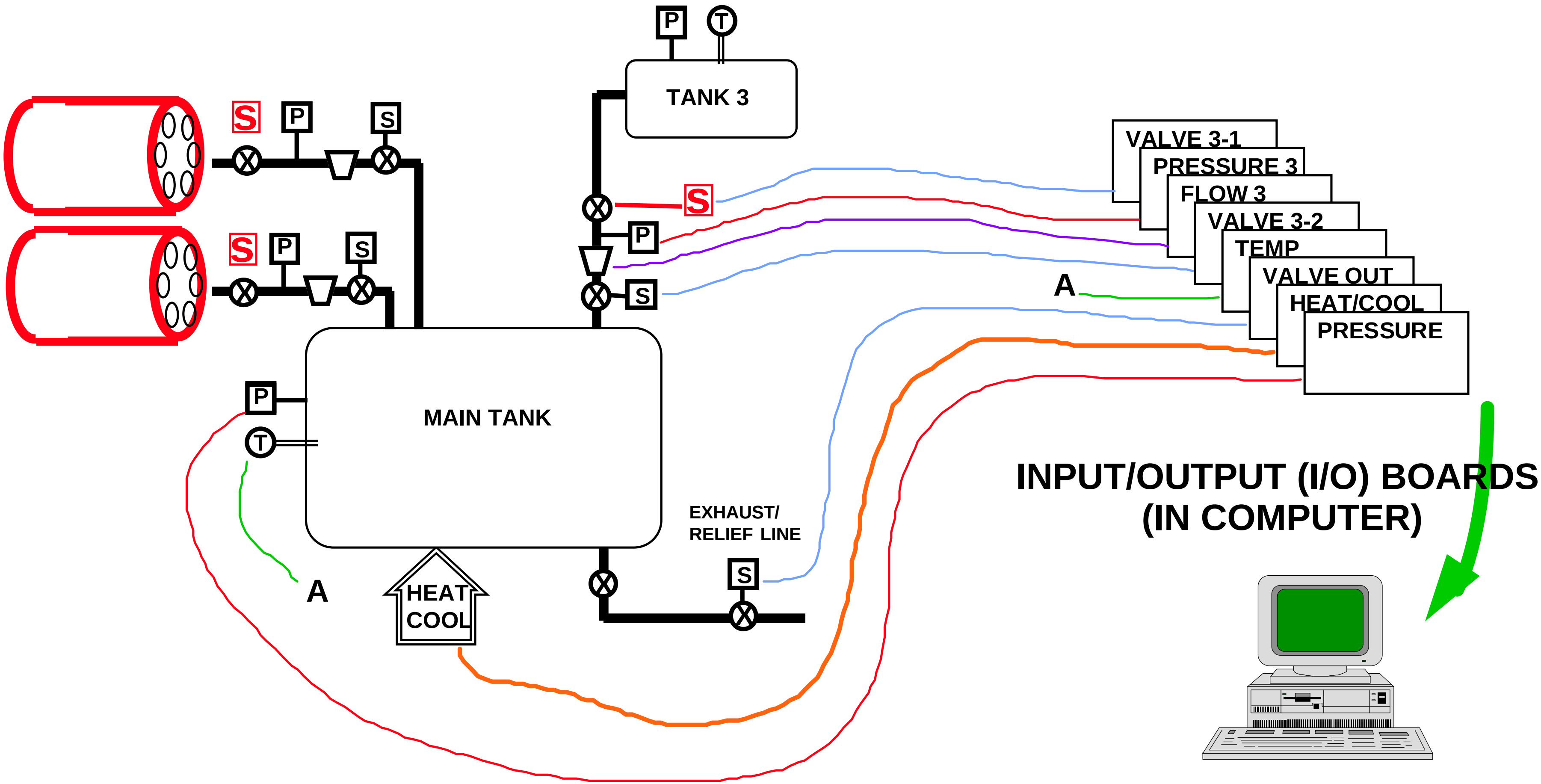


Remove Individual Controllers

MAIN MECHANICAL SCHEMATIC-*REVISED*



COMPUTER CONTROLLED CONVERSION
Chemical Reaction Microgravity Experiment



INPUT/OUTPUT (I/O) BOARDS
(IN COMPUTER)

Tank Systems 1 & 2
and carousel are
connected to computer
in a similar manner.

F; P14-1 p.4

SW FAILURE MODES AND EFFECTS ANALYSIS (FMEA)

- **WHAT CAN GO WRONG WITH SENSORS?**
- **WHAT CAN GO WRONG WITH THE VALVES?**
- **WHAT CAN GO WRONG WITH CABLING?**
- **WHAT CAN GO WRONG WITH THE HEATING AND COOLING SYSTEM?**
- **WHAT CAN GO WRONG WITH THE FRONT ENDS (the I/O Boards in the computer)?**
- **WHAT CAN GO WRONG WITH THE COMPUTER?**
- **WHAT CAN GO WRONG WITH THE SOFTWARE?**

Failure Modes and Effects Analysis Worksheet

ITEM DESCRIPT	FUNCTION	FAILURE MODE	LOCAL EFFECT	SYSTEM EFFECT	CORRECTIVE ACTION	DETECTION	CRIT

PROBLEM 2: RELIABILITY MODEL

- **Musa time domain model:**

$$n = n_0[1-exp(-Ct/n_0 MTTF_0)]$$

Where:

n = **number of errors at time** t

n_0 = **the inherent number of errors, 400**

t = **program execution time**

$MTTF_0$ = **the MTTF start of testing, 2.0 hours**

C= **testing speed vs. typical run speed, 3**

RELIABILITY MODEL (con't)

Also:

$$MTTF = MTTF_0 \, exp(-Ct / n_0 MTTF_0)$$ **and**

$$R = exp(-t / MTTF)$$

Therefore:

$$\Delta n = n_0 MTTF_0 \; [(1/ MTTF_1)-(1/ MTTF_2)]$$

$$\Delta t = (n_0 MTTF_0 / C) \, ln \, (MTTF_2 / MTTF_1)$$

RELIABILITY MODEL (con't)

A program is estimated to contain, n_0 = 400 errors and initial MTTF, $MTTF_0$ = 2.0 hr. The compression factor is, C= 6. How much testing is needed to reduce the number of errors to 5.

$$(400\text{-}5) = (400 \times 2[(1/2)\text{-}(1/MTTF_2)]$$

$$t = [(400 \times 2)/4) \ln (MTTF_2 /2)$$

$$MTTF_2 = \underline{\hspace{3cm}} \text{ and } \Delta t = \underline{\hspace{4cm}}$$

F; P14-2 p.1

OVERVIEW: DEFINITIONS

- RELIABILITY
  - The probability that an item can perform its intended function for a specified interval under stated conditions.
- QUALITY
  - Conformance to requirements; degree to which a product, function, or process meets the customers' and users' requirements.
- SAFETY (any unreliability is unsafe).
  - Freedom from whatever exposes a person or equipment to potential harm; Also: System reliability when mission success means that no accident is caused by failure of a unit.

SOFTWARE DEFINITIONS

- SOFTWARE RELIABILITY
  - The application of reliability engineering techniques which improve the duration or probability of failure-free performance under stated conditions.
- SOFTWARE QUALITY
  - The totality of features and characteristics of a software product that determines its ability to satisfy given needs or conform to specifications.
- SOFTWARE ENGINEERING
  - The application of engineering problem solving techniques to produce digital systems.

SOFTWARE DEFINITIONS (con’t)

- SOFTWARE SAFETY
  - The application of system safety engineering techniques to software development in order to ensure and verify that software design takes positive measures to enhance the safety of the system and eliminate or control errors which could reduce the safety of the system.

SOFTWARE REQUIREMENTS DEFINITIONS

- **REQUIREMENTS**
  - **Condition or capability needed by a user to solve a problem or achieve an objective.**
- **SPECIFICATION**
  - **Document that prescribes, in a complete precise, verifiable manner the requirements.**
- **REQUIREMENTS SPECIFICATION**
  - **Specification that sets forth the requirements for a system or system component.**

SOFTWARE REQUIREMENTS DEFINITIONS (con't)

- VERIFICATION
  - The process of determining whether or not the products of a given phase of the software development cycle fulfill the requirements established during the previous phase.
- VALIDATION
  - The process of evaluating software at the end of the software development process to ensure compliance with software requirements.